Lecture 16: *Kinematics Transformation: Part 2*

Scribe: *Andrew Zhao - Antariksh Narain - Shihan Lu - Thomas Weaver*

October 21, 2019

1 Jacobian Equation

1.1 Derivation of Jacobian Equation

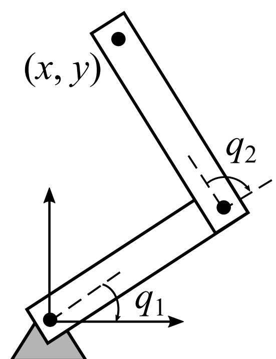


Figure 1: A 2 DOF robotic arm

$$P = Q\left(q\right)$$

$$\frac{\partial P}{\partial t} = \frac{\partial Q\left(q\right)}{\partial q}\frac{\partial q}{\partial t} \quad \Rightarrow \quad \dot{p} = \frac{\partial Q(q)}{\partial q}\dot{q}$$

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} l_1 \cos q_1 + l_2 \cos\left(q_1 + q_2\right) \\ l_1 \sin q_1 + l_2 \sin\left(q_1 + q_2\right) \end{bmatrix} = \phi\left(q_1, q_2\right) = \begin{bmatrix} \phi_1\left(q_1, q_2\right) \\ \phi_2\left(q_1, q_2\right) \end{bmatrix}$$

Jacobian for 2 DOF robot arm:

$$J = \nabla_q Q = \frac{\partial Q(q)}{\partial q} = \begin{bmatrix} \frac{\partial \phi_1}{\partial q_1} & \frac{\partial \phi_1}{\partial q_2} \\ \frac{\partial \phi_2}{\partial q_1} & \frac{\partial \phi_2}{\partial q_2} \end{bmatrix} = \begin{bmatrix} -l_1 \sin q_1 - l_2 \cos(q_1 + q_2) & l_2 \sin(q_1 + q_2) \\ l_1 \cos q_1 + l_2 \cos(q_1 + q_2) & l_2 \cos(q_1 + q_2) \end{bmatrix} \tag{1}$$

1.2 Example

$$l_1 = l_2 = 1;\ q_1 = \frac{\pi}{4};\ q_2 = \frac{\pi}{2}$$

$$J = \begin{bmatrix} -\sqrt{2} & -1/\sqrt{2} \\ 0 & -1/\sqrt{2} \end{bmatrix}$$

using equation: $\dot{p} = J\dot{q}$

$$\dot{p} = \begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} = J\dot{q} = \begin{bmatrix} -\sqrt{2} & \sqrt{2}/2 \\ 0 & -\sqrt{2}/2 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} -\sqrt{2} \\ 0 \end{bmatrix}$$

2 Gradient Descent Method

For inverse kinematic problems, one approach to estimate q by given p is to utilize the gradient descent method to search the "q-space". From the initial configuration, we take a small step in the opposite direction of the gradient. This produces a new configuration and is repeated until the gradient becomes zero. Algorithm 1 defines the basic procedure of the gradient descent method.

Algorithm 1 Gradient Descent

Input: A means to compute the gradient $\nabla U(q)$ **at a point** q
Output: A sequence of points $\{q(0), q(1), \ldots, q(i)\}$

$q(0) = q_{start}$
$i = 0$
while $\nabla U(q(i)) \neq 0$ **do**
    $q(i+1) = q(i) + \alpha(i)\nabla U(q(i))$
    $i = i + 1$

$q(i)$ denotes the value of q at the ith iteration. The final path is $\{q(0), q(1), \ldots, q(i)\}$. The scalar $\alpha(i)$ specifies the step size at iteration i. $\alpha(i)$ should be small enough that the robot does not ”jump into” obstacles, while also being large enough that the algorithm still runs efficiently. $\alpha(i)$ is often chosen on an *ad hoc* or empirical basis. The condition $\nabla U(q(i)) = 0$ is unlikely to be satisfied. Therefore, this condition is usually replaced with $|\nabla U(q(i))| < \epsilon$, where ϵ is sufficiently small, based on the task requirements.

Now we apply the gradient descent method for the inverse kinematic problem.

$$F = \frac{1}{2}(p_d - p)^T(p_d - p) \tag{2}$$

By the gradient descent approach, we have

$$q_{t+1} = q_t - \alpha \left[\frac{\partial F}{\partial q}\right]^T = q_t - \alpha[\nabla_q f]^T \tag{3}$$

$$\nabla_q f = \left[\frac{\partial f}{\partial q_1}, \frac{\partial f}{\partial q_1}, \ldots, \frac{\partial f}{\partial q_n}\right] \tag{4}$$

Since $p = Q(q)$, we rewrite equation (2) as:

$$F = \frac{1}{2}(p_d - Q(q))^T(p_d - Q(q)) \Rightarrow \frac{\partial F}{\partial q} = -\frac{\partial Q(q)^T}{\partial q}(p_d - p) \tag{5}$$

$$q_{t+1} = q_t - \alpha[-\frac{\partial Q(q)^T}{\partial q}(p_d - p)] = q_t + \alpha\frac{\partial Q(q)^T}{\partial q}(p_d - p) \tag{6}$$

Gradient descent expression in IK:

$$q_{t+1} = q_t + \alpha J^T(q)\Delta p \tag{7}$$

3 Jacobian Pseudoinverse

For a redundant manipulator, it usually has a “fat” jacobian matrix which has redundant degrees of freedom to execute a given task. The inverse kinematic problem of this type of manipulator raises interest since it may contain infinite results. To make the gradient descent method on the ”fat” jacobian matrix feasible and efficient, we introduce the concept of Jacobian pseudoinverse.

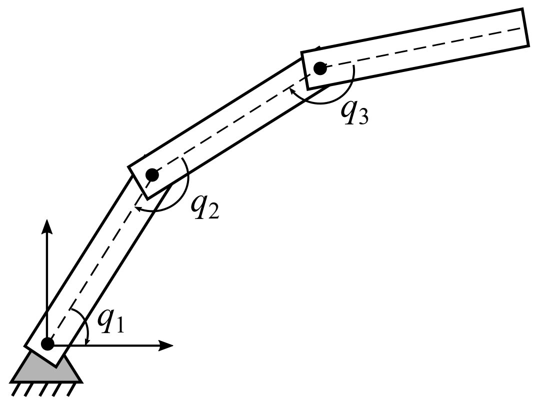


Figure 2: A diagram of a 3-link manipulator

$$J = \nabla\phi(q) = \begin{bmatrix} \frac{\partial\phi_1}{\partial q_1} & \frac{\partial\phi_1}{\partial q_2} & \frac{\partial\phi_1}{\partial q_3} \\ \frac{\partial\phi_2}{\partial q_1} & \frac{\partial\phi_2}{\partial q_2} & \frac{\partial\phi_2}{\partial q_3} \end{bmatrix} \tag{8}$$

$$\min \ \frac{1}{2}\Delta q^T \Delta q \tag{9}$$

$$s.t. \quad \Delta p = J\Delta q \tag{10}$$

where $\Delta p = p_d - p$ and $\Delta q = q - q_0$.

By the Lagrangian multiplier from numerical optimization, we get

$$F(\Delta q, \lambda) = \frac{1}{2}\Delta q^T \Delta q + \lambda(\Delta p - J\Delta q) \tag{11}$$

To solve this optimization problem, we need to let the partial derivative of F in terms of Δq and λ equals to zero as following.

$$\begin{cases} \frac{\partial F}{\partial \Delta q} = 0 \Rightarrow \Delta q - J^T\lambda = 0 \Rightarrow \Delta q = J^T\lambda \\ \frac{\partial F}{\partial \lambda} = 0 \Rightarrow \Delta p - J\Delta q = 0 \Rightarrow \Delta p = J\Delta q \end{cases} \tag{12}$$

By simplifying the equations above, we have

$$\begin{cases} J\Delta q = JJ^T\lambda \\ \Delta p = J\Delta q \end{cases} \tag{13}$$

$$\Delta p = JJ^T\lambda \tag{14}$$
$$\lambda = (JJ^T)^{-1}\Delta p \tag{15}$$

Then put the equation (15) into equation (12), we have

$$\Delta q = J^T(JJ^T)^{-1}\Delta p \tag{16}$$

Here, $J^T(JJ^T)^{-1}$ is denoted as $J^\#$ and is called **pseudoinverse of J**. Therefore, the gradient descent equation can be rewritten as following.

$$q_{t+1} = q_t + \alpha J^\# \Delta p \tag{17}$$

4 When Jacobian is Zero

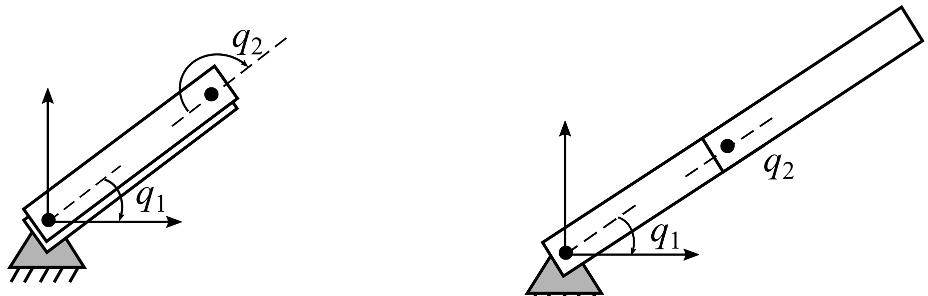
$$
\begin{gathered}
|J| = 0 \\
-l_1 \sin q_1 \cdot l_2 \cos(q_1+q_2) - {l_2}^2 \cdot \sin(q_1+q_2)\cos(q_1+q_2) \\
+l_2 \cdot \sin(q_1+q_2)_1 \cdot \cos(q_1) + {l_2}^2 \cdot \sin(q_1+q_2)\cdot\cos(q_1+q_2) = 0 \\
l_1l_2(\sin q_1 \cos q_2 \cos q_1 + \cos q_1 \sin q_2 \cos q_1 - \sin q_1 \cos q_2 \cos q_1 + \sin^2 q_1 \sin q_2) = 0 \\
l_1l_2(\sin q_2(\cos^2 q_1 + \sin^2 q_1)) = 0 \\
l_1l_2 \sin q_2 = 0 \\
\sin q_2 = 0 \\
q_2 = 0 \;\; or \;\; \pi
\end{gathered}
$$

For example, if $q_2 = 0$,

$$J = \begin{bmatrix} -(l_1+l_2)\sin q_1 & -l_2 \sin q_1 \\ (l_1+l_2)\cos q_1 & l_2\cos q_1 \end{bmatrix}$$

So, the rank of J is 1, which means J is not invertible.

$$\dot{p} = J \cdot \dot{q} = \begin{bmatrix} J_1 & J_2 \end{bmatrix} \begin{bmatrix} \dot{q_1} \\ \dot{q_2} \end{bmatrix}$$



(a) $q_2 = \pi$

(b) $q_2 = 0$

Figure 3: Two singlarity cases of 2 DOF robotic arm

5 Summary

1. Inverse kinematics can be solved by the following methods:
   - Using geometry (not a perfect closed form)
   - Gradient Descent (need many iterations to converge in certain configurations)
   - Gradient Descent with Jacobian pseudo inverse (least effort solution but not conservative)